

CERTIFICATE OF DEPOSIT

I hereby certify that this paper and every paper referred to therein as being enclosed is being electronically deposited with the USPTO via the USPTO EFS portal.

on January 6, 2009 (Date of Deposit – U.S. Pacific Standard Time (PST))

Craig S. Fischer

By

Signature

27662

PATENT & TRADEMARK OFFICE

PATENT

Attorney Docket No.: MCS-057-00

MSFT No.: 155593.01

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE BEFORE THE
BOARD OF PATENT APPEALS AND INTERFERENCES**

In re the Application of: HUANG et al.

Serial No.: 09/772,382

Group Art Unit: 3623

Filed: January 29, 2001

Examiner: J. G. STERRETT

For: **SYSTEM AND METHOD FOR HIGH-DENSITY
INTERACTIVE VOTING USING A COMPUTER NETWORK**

APPEAL BRIEF UNDER 37 CFR 41.37

I. REAL PARTY IN INTEREST

Microsoft Corporation owns the subject application in its entirety.

II. RELATED APPEALS AND INTERFERENCES

There are no known related appeals or interferences.

III. STATUS OF CLAIMS

1. Claims 1-7, 9, 11-15, and 18-32 represent all claims currently pending in the application.
2. Claims 1-7, 9, 11-15, and 18-32 are rejected.
3. Claims 8, 10, 16, and 17 have been canceled.
4. The rejection of claims 1-7, 9, 11-15, and 18-32 is hereby appealed.

IV. STATUS OF AMENDMENTS

A Final Office Action was issued on April 21, 2008 which rejected claims 1-7, 9, 11-15, and 18-32. A Notice of Appeal was filed on July 21, 2008. There were no amendments filed subsequent to the final rejection dated April 21, 2008.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

The pending patent application includes 4 independent claims: claims 1, 14, 22, and 26. A summary of the subject matter recited in claims 1, 14, 22, and 26 is provided below.

a. Subject Matter of Independent Claim 1:

In general, the subject matter of independent claim 1 relates to a method for facilitating interactive voting over the Internet during a corresponding live television broadcast event (specification, page 3, lines 12-15). The method includes presenting a survey question and a plurality of responses to voters viewing the live television broadcast event (specification, page 1, lines 30-31). The method directs voters to cast votes over the Internet at a web site of a sponsor of the live television broadcast event (specification, page 2, lines 17-20). Each of the votes is transmitted over the Internet to a server of the web site (specification, page 2, lines 5-7), and the web site server receives raw votes from the voters over the Internet at the web site server in response to the survey question (specification, page 2, lines 7-9).

The method provides a Live Event Object residing on the server (specification, page 11, lines 1-2), and that maintains persistent connections between the Live Event Object and a database (specification, page 5, lines 29-31). The method also caches the raw votes received from the voters in a memory cache of the Live Event Object for a

predefined time interval the raw votes having never been written in a database (specification, page 3, lines 34-35 to page 4, line 1). In particular, the LEO “caches the votes received from voters and performs batch vote processing before sending the batch results to the Live Event Database” (specification, page 3, lines 34-35 to page 4, line 1).

The method also tabulates as a batch in the memory cache the cached raw votes accumulated over the predefined time interval to generate intermediate voting results (specification, page 4, lines 3-6). The votes are cached and tabulated in the Live Event Object prior to writing in the database (specification, page 11, lines 16-18). The method then writes the intermediate voting results and each raw vote accumulated over the predefined time interval to the database at the predefined interval only after each raw vote received has been cached and tabulated as a batch in the memory cache (specification, page 12, lines 23-24). The final voting result to the survey question is computed in real time by continuously tallying each of the intermediate voting results written in the database (specification, page 6, lines 3-5), and the final voting results are presented to viewers on the live television broadcast event prior to its conclusion (specification, page 12, lines 31-33 and page 2, lines 10-13).

b. Subject Matter of Independent Claim 14:

Generally, the subject matter of independent claim 14 relates to an interactive voting system using a computer network (specification, page 5, lines 12-14). The system includes a server in communication with the computer network for receiving votes from a plurality of voters in response to a polling question presented to the voters during a live broadcast event that directs the plurality of voters to respond to the polling question by visiting a web site (specification, page 9, lines 17-25). The system also includes an object residing in a memory cache on the server for caching raw votes received from the voters during a predefined time interval, the raw votes having never been written to a database, and summing as a batch the raw votes accumulated during that predefined time interval to compute an intermediate voting result (specification, page 3, lines 34-35 to page 4, line 1; page 13, lines 7-9). The object is a non-relational object. The system also includes a database having a connection with the object that receives and writes the intermediate voting result and each raw vote received during the predefined time interval to the database

at the predefined time interval only after each raw vote received has been cached and tabulated as a batch in the memory cache (specification, page 12, lines 23-24). A plurality of intermediate voting results for different time intervals are generated (specification, page 11, lines 24-29). The system also includes a final voting result tabulated in real time by summing each of the plurality of intermediate voting results and presented during the live broadcast event (specification, page 11, lines 28-29; and page 3, lines 12-15). The votes are cached and summed in the object prior to writing in the database (specification, page 13, lines 5-9).

c. Subject Matter of Independent Claim 22:

In general, the subject matter of independent claim 22 relates to an interactive voting system that uses a computer network to process voting data in response to a survey question asked during a live television broadcast (specification, page 3, lines 12-15). The system includes a Live Event Vote Server in communication with the computer network and accessible at a web site of a sponsor of the live television broadcast (specification, page 9, lines 18-19; page 9, lines 10-16). The system also includes a Live Event Object residing in a memory cache on a Live Event Vote Server (specification, page 11, lines 1-2). The Live Event Object receives and caches voting data over a predefined time interval from a client in communication with the computer network, where the voting data has never been written in a database and the voting data represents responses to the survey questions given by viewers of the live television broadcast after having visited the sponsor's web site (specification, page 3, lines 34-35 to page 4, line 1; page 2, lines 17-20). The system tabulates as a batch the cached voting data accumulated over the predefined time interval to generate an intermediate voting result (specification, page 4, lines 3-6), and writes the intermediate voting results and each raw vote accumulated over the predefined time interval to the database which is a Live Event Database through persistent connections between the Live Event Object and the Live Event Database only after each raw vote received has been cached and tabulated as a batch in the memory cache (specification, page 3, lines 34-35 to page 4, line 1; page 13, lines 7-9). The intermediate voting result is used to compute a final voting result in real-time and the final voting result is presented to television viewers during the live television broadcast, wherein

the voting data is cached and tabulated in the Live Event Object prior to writing to the Live Event Database (specification, page 11, lines 28-29; and page 3, lines 12-15).

d. Subject Matter of Independent Claim 26:

In general, the subject matter of independent claim 26 relates to, in a computer network having a plurality of clients and a server, a computer-implemented method for providing interactive voting over the Internet during a live television broadcast (specification, page 3, lines 12-15). The method includes presenting a survey question and a number of responses to voters viewing the live television broadcast (specification, page 1, lines 30-31), and directing voters viewing the live television broadcast to cast a raw vote for one or more of the responses by using at least one of the plurality of clients to visit a web site of a sponsor of the live television broadcast (specification, page 2, lines 17-20). The method also includes transmitting votes submitted by the voters using the plurality of clients over the Internet to the server located at the sponsor's web site (specification, page 2, lines 5-7), providing an object resident in memory on the server that contains procedures and instructions for manipulating the raw votes (specification, page 11, lines 1-2), and accumulating the raw votes in the server memory in a memory cache during a predefined time interval, where the raw votes have never been written in a database (specification, page 3, lines 34-35 to page 4, line 1).

The method also includes tabulating as a batch in the memory cache the accumulated cached raw votes at the end of the predefined time interval to generate an intermediate voting result (specification, page 4, lines 3-6), and the votes are cached and tabulated prior to writing in the database (specification, page 11, lines 16-18). The method further includes writing the intermediate voting result and each raw vote accumulated over the predefined time interval to the database at the end of the predefined time interval only after each raw vote received has been cached and tabulated as a batch in the memory cache (specification, page 12, lines 23-24), establishing and maintaining a persistent connection between the object and the database to facilitate writing of the intermediate voting results (specification, page 5, lines 29-31), repeating the accumulation of votes and the writing of intermediate results to the database to obtain a plurality of intermediate results (specification, page 4, lines 3-6), tabulating the plurality of intermediate results to

obtain a final voting result in real time (specification, page 11, lines 28-29; and page 3, lines 12-15), and presenting the final voting results within time constraints of the live television broadcast (specification, page 11, lines 28-29; and page 3, lines 12-15).

VI. GROUND S OF REJECTION TO BE REVIEWED ON APPEAL

The Office Action rejected claims 1-7, 9, 11-15, and 18-32 under 35 USC §103(a) as being obvious over Bayer et al. (U.S. Patent No. 6,311,190) in view of Oracle 8i and further in view of Blumberg (U.S. Patent No. 6,240,415). Oracle 8i is described in two papers: “Programming Environments for Oracle Objects”, pp. 1-18 (hereinafter referred to as Reference “A”), and “Programmatic Environments”, pp. 1-27 (hereinafter referred to as Reference “B”). The rejection of independent claim 1, dependent claims 2-7, 9, and 11-13, independent claim 14, dependent claims 15, and 18-21, independent claim 22, dependent claims 23-25, and independent claim 26, and dependent claims 27-32 is appealed.

VII. ARGUMENT

The following arguments present the rationale for the patentability of independent claims 1, 14, 22, and 26. The specific rejections of dependent claims 2-7, 9, 11-13, 15, 18-21, 23-25, and 27-32 are not argued separately, as it is believed that the patentability of their independent parent claims 1, 14, 22, and 26 negates the specific rejections advanced with respect to these dependent claims.

a. Rejection of Claims 1-7, 9, 11-15, and 18-32 under 35 USC §103(a) over Bayer et al. in view of Oracle 8i and further in view of Blumberg

a.1 Independent Claims 1, 14, 22, and 26:

The Appellants had previously amended independent claims 1, 14, 22, and 26 and argued that neither Bayer et al., the Oracle 8i papers, nor Blumberg, either explicitly or implicitly, discloses the material claimed features of:

1. (As recited in independent claim 1): “caching the raw votes received from the voters in a memory cache of the Live Event Object for a predefined time interval, the raw votes having never been written in a database”; and

“writing the intermediate voting results and each raw vote accumulated over the predefined time interval to the database at the predefined interval only after each raw vote received has been cached and tabulated as a batch in the memory cache.”

2. (As recited in independent claim 14): “an object residing in a memory cache on the server for caching raw votes received from the voters during a predefined time interval, the raw votes having never been written to a database”; and
“a database having a connection with the object that receives and writes the intermediate voting result and each raw vote received during the predefined time interval to the database at the predefined time interval only after each raw vote received has been cached and tabulated as a batch in the memory cache.”
3. (As recited in independent claim 22): “a Live Event Object residing in a memory cache on a Live Event Vote Server, the Live Event Object receiving and caching voting data over a predefined time interval from a client in communication with the computer network, the voting data having never been written in a database”; and
“writing the intermediate voting results and each raw vote accumulated over the predefined time interval to the database which is a Live Event Database through persistent connections between the Live Event Object and the Live Event Database only after each raw vote received has been cached and tabulated as a batch in the memory cache.”
4. (As recited in independent claim 26): “accumulating the raw votes in the server memory in a memory cache during a predefined time interval, the raw votes having never been written in a database”; and
“writing the intermediate voting result and each raw vote accumulated over the predefined time interval to the database at the end of the

predefined time interval only after each raw vote received has been cached and tabulated as a batch in the memory cache.”

Independent claims 1, 14, 22, and 26 were rejected under 35 U.S.C. §103(a) based on the rationale that the combination of Bayer et al., Oracle 8i, and Blumberg teaches all the features of this claim. In particular, the Examiner stated that Bayer et al. teach the majority of the features recited in independent claims 1, 14, 22, and 26, except that “Bayer does not teach . . . caching the votes received and tabulating as a batch in a memory cache using the object.” However, the Examiner contended that Oracle 8i teaches these features.

In particular, the Examiner stated that the feature of “caching the raw votes received from the voters in a memory cache, where the raw votes have never been written in a database” is disclosed in Reference “A,” page 2, paragraph 3, line 1, by “a client side object cache for caching objects in memory.” Moreover, the Examiner stated that in Reference “A,” page 5, paragraph 2, line 1, is taught “when pinning an object, the duration an object is pinned in memory, that is to have computations performed on the object, can be specified by the programmer. That is, the programmer can predefine the intervals in which votes are tabulated in memory.”

Moreover, the Examiner maintained that feature of “writing the intermediate voting results and each raw vote to the database only after each raw vote received has been cached and tabulated as a batch in the memory cache” is disclosed in Reference “A,” page 2, paragraph 3, line 7. In particular, this section of Reference “A” states that “objects in the cache, including those which contain intermediate results as a result of calculations . . . can be written (i.e. flushed) to the database. This means that any calculation including counting (i.e. tabulating of vote results), can be tallied to the database when the predefined interval for updating the cache occurs.”

The Appellants have argued previously that independent claims 1, 14, 22, and 26 recite accumulating in a memory cache the raw votes received from voters over a network (such as the Internet). These raw votes have never been written in a database. In other

words, the raw votes (which are incoming votes cast by a voter over the computer network) are received from the network and then placed in the memory cache without being written to a database.

The high-density interactive voting system and method “caches votes received from voters and performs batch processing prior to sending the batch results to the Live Event Database” (specification, page 11, lines 16-18; emphasis added). The tabulation of raw votes in the memory cache after a certain time interval generates intermediate voting results.

Once these intermediate voting results are generated in the memory cache, only then are both the intermediate voting result and the raw votes written to the database. The votes are cached and tabulated in the memory cache and then written in the database. Up to this time, the raw votes have never been written to a database, having been received from the voter over the network, placed in the memory cache, and tabulated to generate the intermediate voting results.

In contrast, the combination of Bayer et al., the Oracle 8i papers, and Blumberg merely discloses manipulating data that has been loaded into a cache from a database. In other words, the data being manipulated has already been written to a database. The combination merely discloses taking a data object that has been written in a database, loading the data object into a cache, processing the data object, and writing the changes made to the object to the database.

The Oracle 8i papers disclose an Oracle Call Interface (OCI) that “applications can use to manipulate data and schemas in an Oracle database” (Reference A, page 1, first two sentences under the heading “Oracle Call Interface (OCI)). The data has already been written to the database. The OCI supports a “client side object cache for caching object in memory”, and supports “flushing changes made to objects in the client cache to the database” (Reference A, page 2, second paragraph under the heading “Navigational Access”, bullets 1 and 7; emphasis added). Moreover, “database object are loaded into the cache” (Reference A, page 2, second paragraph under the heading “Object Cache”, line 1). In other words, the

data has already been written to the database. Once written to the database, the data can be loaded into a cache and processed, and the changes are written to the database.

In the “Response to Arguments” section of the Final Office Action, the Examiner noted that “the applicant cites individual passages from the cited Oracle 8i reference that suggests the teaching is opposite from the claimed invention. To this the Examiner would point out that the fact that Oracle teaches pulling a copy from a database to write to it does not teach away from the claimed invention, rather it supports the teaching of tabulating information in a cache prior to writing it to a database. One of ordinary skill in the art would recognize that (1) performing operations in a cache is faster than performing those operations in a database (2) the operations in a cache require a mapping of where the data would go in the database once those operations in the cache are performed.”

The Examiner goes on to say that a “cache is nothing more than a different type of memory where operations can be performed and results stored in a much faster manner than is would be performed in the database memory.” The Examiner also says that “[O]ne of ordinary skill in the art would recognize that the writing of data from the database into the cache would be the starting point – the objects, i.e. the data, would then be modified while in the cache prior to writing the final result to the database.” It is the Appellants’ understanding that the Examiner is arguing that irrespective of where the data is obtained, both Oracle 8i and the Appellants’ invention recited in independent claims 1, 14, 22, and 26 recite first manipulating data in a memory cache and then writing the changes to the data to a database.

While this is true in part, the Appellants’ claimed invention as recited in amended claims 1, 14, 22, and 26 also differs in at least two important ways. First, the data that is in the memory cache has never been written to a database. This data is the raw votes received over the network from voters and sent to the memory cache for processing. This processing includes tabulation, where all the received raw votes during a certain time interval are summed to generate an intermediate voting result. Second, both the intermediate voting result and the raw votes then are written to the database. This is the first time that both the intermediate voting results and the raw votes are written to a database. Up until this time they have resided in the memory cache.

The Appellants, therefore, submit that obviousness cannot be established since the combination of Bayer et al., Oracle 8i, and Blumberg fails to teach, disclose, suggest or provide any motivation for the Appellants' material claimed features recited in claims 1, 14, 22, and 26, as outlined above. In addition to explicitly lacking these features, the combination of Bayer et al., Oracle 8i, and Blumberg also fails to implicitly disclose, suggest, or provide explicit or implicit motivation for these features. Therefore, the Appellants maintain that no prima facie case of obviousness can be established due to the lack of some teaching, suggestion, or incentive supporting the combination. The Appellants, therefore, respectfully request reexamination, reconsideration and withdrawal of the rejection of claims 1-7, 9, 11-15, and 18-32 under 35 USC §103(a) as being obvious over Bayer et al. in view of Oracle 8i and further in view of Blumberg.

VIII. CLAIMS APPENDIX

The following claims 1-7, 9, 11-15, and 18-32 represent a complete copy of all claims involved in the appeal of the above-referenced patent application and are provided in accordance with the requirements of 37 C.F.R. § 1.192(c)(7).

Listing of Claims:

1. (Rejected) A method for facilitating interactive voting over the Internet during a corresponding live television broadcast event, comprising:
 - presenting a survey question and a plurality of responses to voters viewing the live television broadcast event;
 - directing the voters to cast votes over the Internet at a web site of a sponsor of the live television broadcast event;
 - transmitting each of the votes over the Internet to a server of the web site;
 - receiving raw votes from the voters over the Internet at the web site server in response to the survey question;
 - providing a Live Event Object residing on the server that maintains persistent connections between the Live Event Object and a database;
 - caching the raw votes received from the voters in a memory cache of the Live Event Object for a predefined time interval, the raw votes having never been written in a database;
 - tabulating as a batch in the memory cache the cached raw votes accumulated over the predefined time interval to generate intermediate voting results, wherein the votes are cached and tabulated in the Live Event Object prior to writing in the database;
 - writing the intermediate voting results and each raw vote accumulated over the predefined time interval to the database at the predefined interval only after each raw vote received has been cached and tabulated as a batch in the memory cache;
 - computing in real time a final voting result to the survey question by continuously tallying each of the intermediate voting results written in the database; and
 - presenting the final voting results to viewers on the live television broadcast event prior to its conclusion.

2. (Rejected) A computer-readable medium having computer-executable instructions for performing the method recited in claim 1.
3. (Rejected) The method as set forth in claim 1, wherein the Live Event Object is resident in computer memory on the server.
4. (Rejected) The method as set forth in claim 1, wherein the Live Event Object establishes and maintains at least three persistent connections.
5. (Rejected) The method as set forth in claim 4, wherein the persistent connections include a raw vote cast by each of the voters.
6. (Rejected) The method as set forth in claim 4, wherein the persistent connections include current voting results obtained using the cached votes.
7. (Rejected) The method as set forth in claim 4, wherein the interactive voting is in response to the survey question asked during a live event and the persistent connections include a definition of the live event.
8. (Canceled)
9. (Rejected) The method as set forth in claim 1, further comprising tabulating the intermediate voting results to compute final voting results.
10. (Canceled)
11. (Rejected) The method as set forth in claim 1, further comprising creating the survey question.
12. (Rejected) The method as set forth in claim 11, further comprising defining an event in which the survey question is asked and checking a validity of the survey question and the event definition to ensure accuracy.

13. (Rejected) The method as set forth in claim 11, further comprising determining whether there has been a new survey question created and, if so, then updating the database.

14. (Rejected) An interactive voting system using a computer network, comprising:

a server in communication with the computer network for receiving votes from a plurality of voters in response to a polling question presented to the voters during a live broadcast event that directs the plurality of voters to respond to the polling question by visiting a web site;

an object residing in a memory cache on the server for caching raw votes received from the voters during a predefined time interval, the raw votes having never been written to a database, and summing as a batch the raw votes accumulated during that predefined time interval to compute an intermediate voting result, wherein the object is a non-relational object;

a database having a connection with the object that receives and writes the intermediate voting result and each raw vote received during the predefined time interval to the database at the predefined time interval only after each raw vote received has been cached and tabulated as a batch in the memory cache, such that a plurality of intermediate voting results for different time intervals are generated; and

a final voting result tabulated in real time by summing each of the plurality of intermediate voting results and presented during the live broadcast event;

wherein the votes are cached and summed in the object prior to writing in the database.

15. (Rejected) The interactive voting system as set forth in claim 14, wherein the object is a Live Event Object containing at least some of the voting data as well as procedures and instructions for manipulating at least some of the voting data.

16. (Canceled)

17. (Canceled)

18. (Rejected) The interactive voting system as set forth in claim 14, further comprising a persistent connection between the object and the database that is established and maintained by the object.

19. (Rejected) The interactive voting system as set forth in claim 18, wherein the persistent connection further comprises at least three persistent connections.

20. (Rejected) The interactive voting system as set forth in claim 14, further comprising an authoring system that enables a user to define an event and create polling questions associated with the event for distribution to the voters, wherein the authoring system is located at broadcast studios where the live broadcast event is occurring.

21. (Rejected) The interactive voting system as set forth in claim 20, wherein the authoring system further comprises a staging component that copies the event definition and polling questions to the database.

22. (Rejected) An interactive voting system that uses a computer network to process voting data in response to a survey question asked during a live television broadcast, comprising a Live Event Vote Server in communication with the computer network and accessible at a web site of a sponsor of the live television broadcast, a Live Event Object residing in a memory cache on a Live Event Vote Server, the Live Event Object receiving and caching voting data over a predefined time interval from a client in communication with the computer network, the voting data having never been written in a database, the voting data representing responses to the survey questions given by viewers of the live television broadcast after having visited the sponsor's web site, tabulating as a batch the cached voting data accumulated over the predefined time interval to generate an intermediate voting result, and writing the intermediate voting results and each raw vote accumulated over the predefined time interval to the database which is a Live Event Database through persistent connections between the Live Event Object and the Live Event Database only after each raw vote received has been cached and tabulated as a

batch in the memory cache, such that the intermediate voting result is used to compute a final voting result in real-time and the final voting result is presented to television viewers during the live television broadcast, wherein the voting data is cached and tabulated in the Live Event Object prior to writing to the Live Event Database.

23. (Rejected) The interactive voting system as set forth in claim 22, further comprising a vote cache that receives and caches at least some of the voting data from the Live Event Object.

24. (Rejected) The interactive voting system as set forth in claim 23, further comprising a Live Event Vote Processor that tabulates the cached voting data from the vote cache to generate the intermediate voting results.

25. (Rejected) The interactive voting system as set forth in claim 24, wherein the Live Event Vote Processor tabulates the intermediate voting results to compute a final voting result in real time.

26. (Rejected) In a computer network having a plurality of clients and a server, a computer-implemented method for providing interactive voting over the Internet during a live television broadcast, comprising:

presenting a survey question and a number of responses to voters viewing the live television broadcast;

directing voters viewing the live television broadcast to cast a raw vote for one or more of the responses by using at least one of the plurality of clients to visit a web site of a sponsor of the live television broadcast;

transmitting votes submitted by the voters using the plurality of clients over the Internet to the server located at the sponsor's web site;

providing an object resident in memory on the server that contains procedures and instructions for manipulating the raw votes;

accumulating the raw votes in the server memory in a memory cache during a predefined time interval, the raw votes having never been written in a database;

tabulating as a batch in the memory cache the accumulated cached raw

votes at the end of the predefined time interval to generate an intermediate voting result, wherein the votes are cached and tabulated prior to writing in the database;

writing the intermediate voting result and each raw vote accumulated over the predefined time interval to the database at the end of the predefined time interval only after each raw vote received has been cached and tabulated as a batch in the memory cache;

establishing and maintaining a persistent connection between the object and the database to facilitate writing of the intermediate voting results;

repeating the accumulation of votes and the writing of intermediate results to the database to obtain a plurality of intermediate results;

tabulating the plurality of intermediate results to obtain a final voting result in real time; and

presenting the final voting results within time constraints of the live television broadcast.

27. (Rejected) The computer-implemented method as set forth in claim 26, wherein the persistent connection comprises at least three persistent connections.

28. (Rejected) The computer-implemented method as set forth in claim 26, further comprising caching the cached voting data in a vote cache.

29. (Rejected) The method as set forth in claim 1, further comprising writing each of the received votes to the database to allow cross-tabulation of demographic data.

30. (Rejected) The method as set forth in claim 1, wherein the predefined time interval is approximately fifteen seconds.

31. (Rejected) The method as set forth in claim 1, further comprising:
tabulating in memory a plurality of the intermediate voting results written to the database such that the final voting results are updated; and
writing the updated final voting results to the database.

32. (Rejected) The method as set forth in claim 31, further comprising updating the final voting results approximately every ten seconds.

IX. EVIDENCE APPENDIX

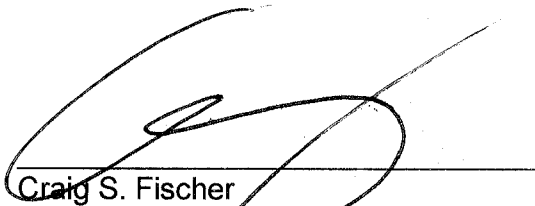
NONE

X. RELATED PROCEEDINGS APPENDIX

NONE

Respectfully submitted,

LYON & HARR, L.L.P.
300 Esplanade Drive, Suite 800
Oxnard, California 93036
(805) 278-8855



Craig S. Fischer
Registration No. 42,535
Attorney for Appellants